

Обучение машинного перевода без параллельных текстов*

Скиднов Е. А. Бахтеев¹ О. Ю. Стрижов² В. В.

¹Московский физико-технический институт

²Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН

Данная задача посвящена исследованию алгоритма обучения машинного перевода без параллельных предложений. Использование параллельных текстов для задачи машинного перевода требует слишком большой базы предложений всех переводимых языков, что является ресурсоемкой задачей для некоторых пар непохожих языков. Особенностью исследуемого алгоритма является то, что для перевода используется кодирование и декодирование текста во внутреннем представлении. Данный алгоритм использует 2 модели нейронной сети Seq2Seq для перевода с одного языка на другой и обратно. Цель данного исследования заключается в том, чтобы сделать векторы скрытых пространств этих двух моделей как можно более близкими. Для демонстрации работоспособности метода используется вычислительный эксперимент машинного перевода между двумя похожими языками: русским и украинским.

Ключевые слова: *машинный перевод, нейросеть, Seq2Seq.*

1 Введение

Благодаря недавним достижениям в области глубокого обучения и наличию крупномасштабных параллельных корпусов, машинный перевод достиг впечатляющей производительности на нескольких языковых парах. Тем не менее, эти модели работают очень хорошо, только если они снабжены огромным количеством параллельных данных в порядке миллионов параллельных предложений. К сожалению, параллельные корпуса стоят дорого [?], поскольку они требуют специализированного опыта и часто не существуют для языков с низким уровнем ресурсов.

Есть несколько подходов к построению оптимального метода обучения. Предлагается использовать рекуррентные нейронные сети с короткой и долгой памятью и нейронные сети, в которых реализовано внимание. В других методах используются нейронные сети, которые осуществляют перевод в два этапа. Такой метода называется Seq2Seq [?].

Данная работа посвящена последнему методу последовательного перевода. Предлагается с помощью первой рекуррентной нейронной сети, основанной на долгой памяти перевести входящую последовательность в вектор, а с помощью второй перевести этот вектор в выходную последовательность на нужном нам языке [1]. Данный метод позволяет гораздо быстрее обучить нейронную сеть переводу с одного языка на другой, в связи с использованием ей предыдущего опыта и наличием у нее памяти и внимания. Проверка и анализ метода проводятся с помощью алгоритма BLEU (Bilingual evaluation understudy) для проверки качества текста, переведенного с одного языка на другой на паре языков русский-украинский.

*Работа выполнена при финансовой поддержке РФФИ, проект № 00-00-00000. Научный руководитель: Стрижов В. В. Авторы: А. В. Грабовой, О. Ю. Бахтеев, В. В. Стрижов, Eric Gaussier, координатор Малиновский Г. С. Консультант: Бахтеев О. Ю.

2 Постановка задачи

Во время обучения нет параллельных пар предложений. Предполагаем, что нам подойдет модель, отображающая предложения из обоих языков в одно общее векторное пространство.

Мы будем использовать модель, в которой используются главных юнита: encoder f и decoder g . f и g в нашей конкретной модели - две рекуррентные нейронные сети. Задача f - отображать предложения в латентное пространство (сразу для двух языков) и g - отображать из латентного пространства в предложения (первого языка и второго соответственно). Введем обозначения: $D^{src} = [s_1^{src}, \dots, s_{m_{src}}^{src}]$, $D^{tgt} = [s_1^{tgt}, \dots, s_{m_{tgt}}^{tgt}]$.

Для реализации этого метода определим функционалы, которые будут минимизироваться. Чтобы модель не обучилась возвращать в конце цикла исходные данные, необходимо зашумить исходные предложения. Пусть $\sigma(x)$ - результат наложения шума на слово x . Оптимизировать будем следующую функцию:

$$L_{AE} = ||d(e(\sigma(x))) - x||^2$$

Пусть дана какая-то модель слабого перевода \hat{g} . На втором шаге функция потерь будет иметь вид:

$$L_{TR} = ||d(e(\hat{g}(e(x)))) - x||^2$$

Пусть дана модель D , различающая скрытые представления векторов предложений из двух языков. На последнем шаге оптимизируем дискримантор, чтобы он различал представления векторов разных языков в скрытом пространстве:

$$L_{ADV} = \log p(\text{lang} = \text{src} | \text{Encoder}(x)) + \log p(\text{lang} = \text{tgt} | \text{Encoder}(y))$$

В итоге нужно минимизировать следующую функцию:

$$L = a * L_{AE} + b * L_{TR} + c * L_{ADV} \longrightarrow \min$$

здесь a, b, c - калибруемые гиперпараметры.

2.1 Гипотеза

Пусть \mathcal{S}^l - множество всех предложений на языке $l \in \{\text{src}, \text{tgt}\}$. Выдвигается гипотеза о том, что существует единое латентное пространство \mathcal{L} и отображения $f^l : \mathcal{S}^l \rightarrow \mathcal{L}$ и $g^l : \mathcal{L} \rightarrow \mathcal{S}^l$ такие, что

- для $\forall l_1, l_2 \in \{\text{src}, \text{tgt}\}$ и $\forall s^{l_1} \in \mathcal{S}^{l_1}$ $g^{l_2}(f^{l_1}(s^{l_1}))$ совпадает с s^{l_1} , если l_1 совпадает с l_2 , и является корректным переводом s^{l_1} , если l_1 и l_2 различаются
- распределения образов $f^{\text{src}}(\mathcal{S}^{\text{src}})$ и $f^{\text{tgt}}(\mathcal{S}^{\text{tgt}})$ совпадают

2.2 Описание метода

Предлагаемый метод заключается в том, чтобы для каждого $l \in \{\text{src}, \text{tgt}\}$ моделировать отображения f^l и g^l кодировщиком \mathbf{f}^l и декодировщиком \mathbf{g}^l соответственно. Таким образом, моделью перевода является композиция $\mathbf{g}^{\text{tgt}} \circ \mathbf{f}^{\text{src}}$.

Оптимизация проводится следующим образом. Функция ошибки содержит три слагаемых, которые соответствуют сделанным предположениям об отображениях f^l и g^l .

2.3 Ошибка восстановления

Для каждого $l \in \{\text{src}, \text{tgt}\}$ рассматривается входное предложение s^l на языке l . Ошибки считаются между s^l и его образом при отображении $\mathbf{g}^l \circ \mathbf{f}^l$.

2.4 Ошибка перевода

Рассматривается предложение \mathbf{s}^{src} без ограничения общности на языке src . В качестве входного предложения используется \mathbf{s}^{tgt} – перевод \mathbf{s}^{src} , полученный с помощью некоторой слабой модели перевода M^0 . Ошибка перевода с языка tgt на язык src считается между \mathbf{s}^{src} и образом \mathbf{s}^{tgt} при отображении $\mathbf{g}^{\text{src}} \circ \mathbf{f}^{\text{tgt}}$. Аналогичным образом считается ошибка перевода с языка src на язык tgt . В качестве M^0 используется пословный переводчик на основе предобученных векторных представлений слов $\mathcal{E}^{\text{src}} = \{\mathbf{x}_i\}_{i=1}^{n_{\text{src}}}$ и $\mathcal{E}^{\text{tgt}} = \{\mathbf{y}_i\}_{i=1}^{n_{\text{tgt}}}$. Переводом слова $\mathbf{x} \in \mathcal{E}^{\text{src}}$ является $\mathbf{y} = \arg \min_{\mathbf{y} \in \mathcal{E}^{\text{tgt}}} \rho(\mathbf{x}, \mathbf{y})$, где ρ – косинусное расстояние (перевод слов языка tgt осуществляется аналогично).

2.5 Штраф за различие распределений

Вводится дискриминатор \mathbf{d} , который решает задачу классификации векторов \mathbf{h}^l латентного пространства \mathcal{L} на классы $\{0, 1\}$: $\mathbf{h}^l \in 0 \Leftrightarrow l = \text{src}$. Векторы \mathbf{h}^l получаются кодировщиком \mathbf{f}^l , в качестве промежуточного результата при отображениях входных предложений. В функцию ошибки добавляется штраф за то, что дискриминатор точно определяет язык входного предложения. Таким образом параметры модели оптимизируются таким образом, чтобы усложнить задачу дискриминатору. В свою очередь параметры дискриминатора оптимизируются параллельно параметрам модели. Соревновательный процесс оптимизации мотивирован желанием добиться сходства распределений латентных векторов для разных языков.

2.6 Детали метода

Введем словарь V^{src} , содержащий проиндексированные слова, встречающиеся в предложениях из $\mathfrak{D}^{\text{src}}$ и переводах предложений из $\mathfrak{D}^{\text{tgt}}$, получаемых с помощью M^0 . Аналогично введем V^{tgt} .

Предложения на разных языках, используемые при оптимизации кодируются с помощью соответствующих словарей. Входные предложения зашумляются преобразованием σ : сначала с некоторой вероятностью q из них удаляется каждое слово, а затем производится случайная перестановка оставшихся слов с условием, что слово не может оказаться дальше чем на k позиций от своей начальной позиции (q, k – гиперпараметры).

Кодировщик \mathbf{f}^l включает в себя слой векторного представления слов \mathbf{e}^l , параметры которого инициализируются векторами \mathcal{E}^l , и рекуррентную нейронную сеть \mathbf{r}^{enc} . Декодировщик \mathbf{g}^l включает в себя \mathbf{e}^l , \mathbf{r}^{dec} и классификатор \mathbf{c}^l , который решает задачу многоклассовой классификации выходов \mathbf{r}^{dec} на классы, соответствующие словам в словаре V^l (отображает выходы \mathbf{r}^{dec} в векторы вероятностей размерности $|V^l|$). Параметры \mathbf{r}^{enc} и \mathbf{r}^{dec} общие для \mathbf{f}^{src} , \mathbf{f}^{tgt} и \mathbf{g}^{src} , \mathbf{g}^{tgt} соответственно.

В качестве меры ошибок классификации используется кросс-энтропия CE . Итоговый вид функции ошибки

$$\begin{aligned}
 L_{\text{tran}} &= w_1 \cdot \sum_{l \in \{\text{src}, \text{tgt}\}} CE(\mathbf{g}^l(\mathbf{f}^l(\sigma(\mathbf{s}^l))), \mathbf{s}^l) + \\
 &+ w_2 \cdot \sum_{l_1 \neq l_2 \in \{\text{src}, \text{tgt}\}} CE(\mathbf{g}^{l_2}(\mathbf{f}^{l_1}(\sigma(\mathbf{s}^{l_1}))), \mathbf{s}^{l_2}) + w_3 \cdot \sum_{l \in \{\text{src}, \text{tgt}\}} CE(\mathbf{d}(\mathbf{f}^l(\sigma(\mathbf{s}^l))), \mathbb{I}[l = \text{src}]) \\
 L_{\text{disc}} &= \sum_{l \in \{\text{src}, \text{tgt}\}} CE(\mathbf{d}(\mathbf{f}^l(\sigma(\mathbf{s}^l))), \mathbb{I}[l = \text{tgt}]).
 \end{aligned}$$

Литература

- [1] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

Поступила в редакцию